

Part I. Regridding technique

[Contents](#) [Previous](#) [Next](#)

Goal: The goal for this tutorial is to show how to find out the gridding of the data and to regrid data from one grid onto another grid.

The strategy:

1) read the CRU data temperature anomaly data called 'temanom'. This is 5 deg gridded data. Get the grid definition from it and also the lat, lon, time and mask definitions that we will use to create the merged data.

2) extract ERA40 data: 2-meter surface air temperature ('tas') and sea surface temperature ('sst'). Regrid the data to the 5 degree grid using the grid from 1).

Lets star with importing all the needed modules, note that we will need the **Regridder** from module **regrid**.

```
# Import modules
import cdms, cdutil, MA, vcs, cdtime
import string, Numeric, time, MV, sys, os
from regrid import Regridder
```

1) read the CRU data:

```
# Open data file
file1 = os.path.join(sys.prefix, 'sample_data/hadcrut2_sample.nc')
a = cdms.open(file1)
print a.listvariable()
```

The output is:

```
['bounds_lon', 'temanom', 'bounds_lat']
```

Read the 'temanom' data and display it's shape

```
# get the start and end time steps
start_time = a.getAxis('time').asComponentTime()[0]
end_time = a.getAxis('time').asComponentTime()[-1]
# get the data
data=a('temanom',time=(start_time, end_time),latitude=(-90,90))
print data.shape
```

(36, 36, 72)

Use 'print data.info()' to see full information about this data. Plot data with default settings.

```
x=vcs.init()
x.setcolormap('default')
x.plot(data)
```

Now get the grid, lat, lon and time definitions from that data.

```
# get grid for regridding
grid1=data.getGrid()
# see how it looks like
print grid1.info()
```

<bound method TransientRectGrid.info of Grid has Python id -0x491e3394.

Gridtype: generic
Grid shape: (36, 72)
Order: yx

```
# get "spatial missing mask"
mask1=data.mask()
# get metadata for final desired data (e.g. latitudes,longitudes,time)
lat=data.getLatitude()
lon=data.getLongitude()
tim=data.getTime()
# close the file
a.close()
```

2) read ERA40 data

```
# Get the ERA40 data for both 2-meter temperature (tas) and ssts (sst)
# on original grid

file2 = os.path.join(sys.prefix, 'sample_data/era40_tas_sample.nc')
b = cdms.open(file2)
# get data
print "reading 'tas' data, please wait..."
tas=b('tas')
print tas.shape
# close the file
b.close()
# now get the 'sst' data
file3 = os.path.join(sys.prefix, 'sample_data/era40_sst_sample.nc')
```

```
b = cdms.open(file3)
# get data
print "reading 'sst' data, please wait..."
sst=b('sst')
b.close()
print sst.shape
```

(35, 160, 320)

Plot 'sst' and 'tas'

```
x=clear()
x.plot(sst)
```

```
y=vcs.init()
y.setcolormap('default')
y.plot(tas)
```

```
# get grid for regridding
grid2=sst.grid()
```

Now lets regrid the 'sst' and 'tas' data to 5-degree grid defined in 1)

```
# setup a regridding function (as: fromgrid, togrid)
regridfunc=Regridder(grid2,grid1)
# create new data with 'togrid' (5-deg) resolution by passing
# the data with 'fromgrid' resolution into the function above
sst_new=regridfunc(sst)
tas_new=regridfunc(tas)
```

Plot 'sst_new' and 'tas_new'

```
x=clear()
x.plot(sst_new)
```

```
y=vcs.init()  
y.setcolormap('default')  
y.plot(tas_new)
```

next: [Part II. Creating sea/land mask and masked data.](#)

[Contents](#) [Previous](#) [Next](#)